Original article

JDMS

Journal of Defense Modeling and Simulation: Applications, Methodology, Technology
XX(X) 1–11

DOI: 10.1177/1548512910391217
http://dms.sagepub.com
SAGE

Unmanned Aircraft Systems as Wingmen

Richard D Garcia[1], Laura Barnes[2] and MaryAnne Fields[3]

Abstract

This paper introduces a method to integrate Unmanned Aircraft Systems (UASs) into a highly functional manned/unmanned team through the design and implementation of 3D distributed formation/flight control algorithms with the goal to act as wingmen for a manned aircraft. The proposed algorithms are designed to increase UAS autonomy, dynamically modify formations, utilize standard operating formations to reduce pilot resistance to integration, and support splinter groups for surveillance and/or as safeguards between potential threats and manned vehicles. The proposed work coordinates UAS members by utilizing artificial potential functions whose values are based on the state of the unmanned and manned assets including the desired formation, obstacles, task assignments, and perceived intentions. The overall unmanned team geometry is controlled using weighted potential fields. Individual UASs utilize fuzzy logic controllers for stability and navigation as well as a fuzzy reasoning engine for predicting the intent of surrounding aircrafts. Approaches are demonstrated in simulation using the commercial simulator X-Plane and controllers designed in Matlab/Simulink. Experiments include staggered trail and right echelon formations as well as splinter group surveillance.

Keywords

unmanned systems, formation control, intent prediction.

1. Introduction

Currently there is myriad of work on Unmanned Aircraft Systems (UASs) performing tasks such as border patrol, fire detection, traffic monitoring, and basic intelligence gathering.[1-5] Like most manufacturing robots, commercial UASs are heavily segregated and typically only operate in areas free from direct human contact. In the specific case of UASs, this typically means segregating airspace and preventing manned vehicles from operating within miles of any UAS. The segregation is necessary because human pilots, unlike UASs, are able to rapidly process a great deal of information which helps them avoid collisions when operating in close proximity to other aircrafts. For this reason, the Federal Aviation Administration is reluctant to allow UASs to fly in commercial US airspace.

Recently, the US Air Force has put forth directives to design algorithms to allow unmanned aircrafts to 'integrate seamlessly' with piloted aircraft.[6] These algorithms should ideally require no more a priori information than that which a human pilot requires to recognize the intent of another aircraft. Furthermore, fewer operators should be able to simultaneously direct swarms of aircraft.[7] In order for UASs to make decisions based upon perceived sensory data and operational context, they must be able to mimic and achieve a level of trust approaching a human piloted aircraft. The commercial sector has also expressed an interested in the application of this technology. FedEx envisions an airfleet of linked drones that can fly in a formation directed by a single piloted aircraft.[6]

The majority of UASs currently deployed throughout the world operate as remotely piloted vehicles. This teleoperated implementation inherently creates a level of diminished capacity for the operators. Many of the senses used by traditional human pilots provide little if any information to UAS pilots. This coupled with diminished situational

[1]Motile Robotics Inc, USA
[2]University of South Florida, USA
[3]Army Research Laboratory, Aberdeen Proving Ground, USA

Corresponding author:
Laura Barnes, University of South Florida, Tampa, FL 33612, USA.
Email: lbarnes3@mail.usf.edu.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **2010**	2. REPORT TYPE	3. DATES COVERED **00-00-2010 to 00-00-2010**
4. TITLE AND SUBTITLE **Unmanned Aircraft Systems As Wingmen**		5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Army Research Laboratory,Aberdeen Proving Ground,MD,21005**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology February 15, 2011

14. ABSTRACT
This paper introduces a method to integrate Unmanned Aircraft Systems (UASs) into a highly functional manned/unmanned team through the design and implementation of 3D distributed formation/flight control algorithms with the goal to act as wingmen for a manned aircraft. The proposed algorithms are designed to increase UAS autonomy dynamically modify formations, utilize standard operating formations to reduce pilot resistance to integration, and support splinter groups for surveillance and/or as safeguards between potential threats and manned vehicles. The proposed work coordinates UAS members by utilizing artificial potential functions whose values are based on the state of the unmanned and manned assets including the desired formation, obstacles, task assignments, and perceived intentions. The overall unmanned team geometry is controlled using weighted potential fields. Individual UASs utilize fuzzy logic controllers for stability and navigation as well as a fuzzy reasoning engine for predicting the intent of surrounding aircrafts. Approaches are demonstrated in simulation using the commercial simulator X-Plane and controllers designed in Matlab/Simulink. Experiments include staggered trail and right echelon formations as well as splinter group surveillance.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**11**	

awareness creates an environment that can cause incorrect assessment of in-flight situations and slow reaction time. Although a great deal of research is being performed to alleviate this problem, such as state-of-the-art human-robot interaction techniques,[8] many researchers believe that increased autonomy will likely minimize the overall effect of this problem.

Increased autonomy for UASs has several desirable characteristics. First, autonomous software agents operating on UASs are not subject to physical or mental fatigue and thus operate indefinitely with the same efficiency and optimality. This is an increasingly important feature due to the desire for persistent surveillance. Second, increased autonomy may allow UAS operators to direct multiple aircraft simultaneously. This aspect, in essence, creates a force multiplier allowing a few well-trained operators to perform missions which require multiple assets to operate simultaneously.

Beyond simply creating a force multiplier, UASs have the appeal of removing the pilots and support crew from dangerous environments. The loss of a UAS due to inclement weather, vehicle failure, or enemy fire is simply a fiscal loss. Given this advantage, an UAS can provide functionality that manned vehicles cannot. For example, an UAS can be deployed to draw enemy fire away from a co-located manned aircraft. There have been some attempts to integrate manned and unmanned system into a seamless team,[9] but more advancement needs to be made before this technology becomes a reality.

Because of the pressing need for manned and unmanned system teaming, the Army Research Laboratory (ARL) created the UASs as Wingmen project. This project is designed to incubate the technology necessary to safely and effectively coordinate manned and unmanned aircraft systems into an efficient team. Although teams of UASs have received a great deal of attention in the research community, the integration of manned vehicles into the team adds several new challenges. These challenges include operating with limited knowledge of other team members' intentions and prioritizing task assignments. This work presents the initial research in support of the UASs as Wingmen project and includes details related to navigation, basic team formations, splinter group surveillance (Section 2), and UAS intent prediction (Section 3).

2. Navigation and Formation Control

As an example scenario, suppose that a manned aircraft, *m*, is being accompanied by a team of *n* UASs. No constraints are placed on the number of unmanned systems. By using a local communication scheme, see Section 2.1, information can be propagated through the networked team with minimal overhead. The *n* UASs have two tasks: (1) accompany the manned vehicle as it performs its mission, and (2) provide reconnaissance of areas of interest as they are identified. The foundation for navigation and formation control consists of potential field and fuzzy logic methods.[10,11] Dynamically weighted potential fields are used to organize the UASs into standard flight formations with the manned vehicle as well as for the splinter group surveillance.[11,12] Obstacle avoidance is detailed in Barnes et al.[11] and briefly described in Section 2.2. Individual UASs, as well as the simulated manned vehicle, are controlled via fuzzy logic. These controllers are discussed in Section 4 and detailed further in Garcia and colleagues.[10, 13,14] Results are presented in Section 5.

2.1 Formation Control

Formation control is performed by allowing each of the *n* UAS team members to set goal locations that correspond with a desired formation. The individual goal selection is based on information provided by a leader vehicle. To reduce the amount of data broadcast to all team members, global information is limited to the position of the manned vehicle. This single global variable allows the UASs to locate the team regardless of their initial position within the world. As UASs approach the vicinity of the manned aircraft, they will dynamically self-order creating individual formation leaders and calculating formation appropriate goal locations. Once an UAS has acquired a formation leader, the single global variable denoting the location of the manned vehicle is no longer needed.

Self-ordering (determining individual formation leaders) is performed via a waterfall type methodology detailed in Algorithm 1. Initially, each UAS attempts to become a direct follower of the manned vehicle. Once an UAS is within close proximity of the manned vehicle, it requests the current formation task. This information includes the desired formation as well as any formation specific details. The local formation is a sub-class of the overall formation and represents a single lead vehicle and its immediate followers. A local formation is considered complete if and only if all immediate follower locations have been filled. If the manned vehicle's local formation is complete the UAS assigns one of the manned vehicle's followers as its leader. This process continues until an incomplete local formation is found, and the UAS can attach itself to a local leader and determine its individual goal location.

Utilizing a local communication scheme, such as in Algorithm 1, has several benefits. First, once a vehicle has paired itself with a leader, manned or unmanned, it only requires local communication with that single leader to join and hold flight formations with the group. This advantage increases the overall scalability of the implementation. Second, the formation can be manipulated and even spliced into multiple sections without needing to inform the entire group. The one obvious disadvantage to this type of communication scheme is that a single link or damaged vehicle can vastly affect the overall success of the formation, but safeguards can be built into the system to overcome this limitation.

Algorithm 1: Find a lead vehicle to follow in the formation

```
function GET_LEADER(leader, self)
{
    dist = CALC_DIST(leader->pos, self->pos)      //calculate distance between two points

    if (dist < leader_threshold)                  //am I close enough to the 'leader' to make a decision
    {
        if (LOCAL_FORM_COMPLETE(leader))          //is the 'leader's' local formation complete?
            self->current_goal = leader->follower //assign 'self' a new leader
        else
        {
            SET_FOLLOWER(leader, self)            //assign 'self' as the follower of the 'leader' vehicle
            SET_LEADER(self, leader)              //set your lead as the 'leader' vehicle
        }
    }
}
```

Individual goal locations are calculated from information received from the local lead vehicles. This information includes the formation type and any formation specific information required for calculating the goal(s). For example, in a right echelon formation, the position of the local lead vehicle is sufficient for determining an appropriate goal location. In a staggered trail formation, local lead vehicle position must be accompanied by vehicle offset (right or left) for the follower to determine an appropriate goal location. For example, if the local lead is staggered to the right then the follower vehicle must assure that it is staggered to the left. Once the appropriate goal location, (x_g, y_g, z_g), is determined, Equation (1) defines the vector of attraction to the goal, consisting of a heading vector and a weight, W_g, which limits the vector between a maximum and minimum velocity, $-v_{\min}$ and $v_{\max}$.

$$\begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = W_g(d_{\text{actual}}, v_{\max}) \begin{pmatrix} x - x_g \\ y - y_g \\ z - z_g \end{pmatrix} \tag{1}$$

The value of W_g is determined from d_{actual} which is the Euclidean distance between the vehicle and the goal location.

Specific formation selection is generally performed ad-hoc and may change several times throughout flight. Formation alterations may be the result of terrain changes, goal alterations, or recently exposed threats. These formation changes must occur quickly and safely. To adhere to this requirement formation information is consistently updated within local formations. Due to the overlapping design of local formations any changes in formation information will be disseminated to all vehicles. This is the result of the natural propagation from leader to follower. Since the manned vehicle is the highest-level leader any changes made by the manned vehicle will propagate to the entire team.

2.2 Obstacle Avoidance

Vector fields weighted with sigmoid functions may be used for obstacle avoidance. This is achieved by creating vectors moving away from obstacle locations (x_{co}, y_{co}, z_{co}). Obstacle avoidance is achieved using Equations (2)–(4):

$$r_{\text{avoid}} = \sqrt{(x - x_{co})^2 + (y - y_{co})^2 + (z - z_{co})^2} \tag{2}$$

$$S_{\text{avoid}}(\alpha_{\text{aviod}}, r_{\text{aviod}}, \Delta R_{\text{aviod}}) = \kappa - \frac{\kappa}{1 + e^{\alpha_{\text{aviod}}(r_{\text{aviod}} - \Delta R_{\text{aviod}})^2}} \tag{3}$$

$$\begin{pmatrix} v_{x\_\text{avoid}} \\ v_{y\_\text{aviod}} \\ v_{z\_\text{aviod}} \end{pmatrix} = S_{\text{aviod}} \begin{pmatrix} x - x_{co} \\ y - y_{co} \\ z - z_{co} \end{pmatrix} \tag{4}$$

The weight function generated by a single obstacle is a sigmoid with maximum value κ shown in (3). The parameter r_{avoid} is the Euclidean distance to a nearby obstacle. The ΔR_{avoid} parameter is the minimum allowable distance between the UASs and obstacles. The α_{avoid} parameter in (3) controls the slope of the S_{avoid} function.

This obstacle avoidance strategy works well for static and predictable slow moving obstacles. In the worst case, the UAS is heading directly towards the obstacle:

$$\begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = -W_g \begin{pmatrix} x - x_{co} \\ y - y_{co} \\ z - z_{co} \end{pmatrix} \tag{5}$$

Adding the two vectors yields

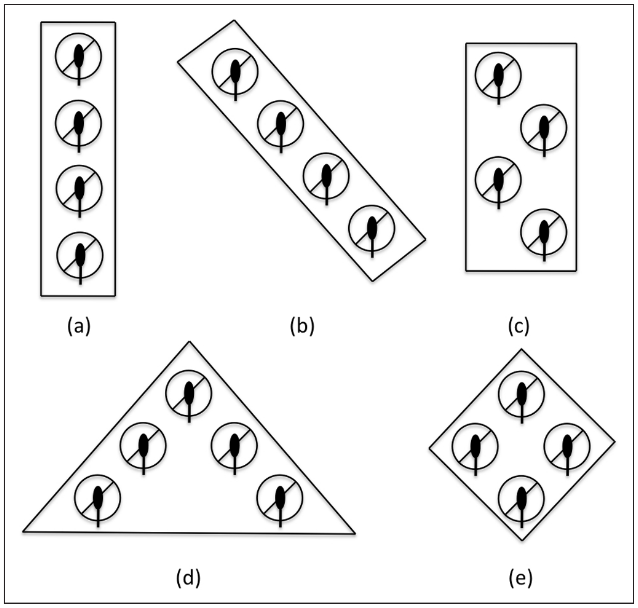

Figure 1. Standard rotary wing formations: (a) trail, (b) right echelon, (c) staggered trail, (d) 'vee', and (e) diamond.

$$\begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} + \begin{pmatrix} v_{x\_\text{avoid}} \\ v_{y\_\text{avoid}} \\ v_{z\_\text{avoid}} \end{pmatrix} = (S_{\text{avoid}} - W_g) \begin{pmatrix} x - x_{co} \\ y - y_{co} \\ z - z_{co} \end{pmatrix} \tag{6}$$

The UAS will move away from the obstacle provided $S_{\text{avoid}} > W_g$ and stop (hover) in a local potential well if $S_{\text{avoid}} = W_g$. Potential wells caused by moving obstacles tend to be transient, stopping the UAS for only an instant. Since stopping is not always possible or desirable, adding a vector that points along the object contour allows the UAS to 'slip' past the obstacle.

2.3 Types of Flight Formations

Aircraft flight formations are designed to coordinate flight, conceal members, and increase safety. Standard flight formations for rotary wing vehicles include trail, staggered trail, echelon, heavy (echelon), diamond and 'vee' shown in Figure 1.[15] Right echelon and staggered trail formations are presented to validate the proposed approach. The addition of formations such as 'vee' and diamond require no modification to the formation control methodology or data structure design since all calculations are local (see Section 2.1). Implementation of these formations would simply require a waterfall notification up the leader chain when a vehicle is added. This would allow the formation to remain balanced as new vehicles enter the team.

2.4 UAS Navigation

To form a complete testing environment the formation control algorithms were integrated with simulated UASs with control algorithms for basic stability and navigation. Each individual UAS is controlled via four distinct fuzzy logic controllers. These four controllers are collectively responsible for controlling the roll, pitch, yaw, and collective. Throttle is controlled by a revolutions per minute (RPM) governor built into the vehicle simulator and designed to maintain a constant head speed throughout the simulation. In depth details of these controllers, including membership functions and complete rule sets, can be found in Garcia.[14] Although Fields et al.[12] does not provide any formal proofs as to the stability of these controllers, the authors have analyzed the performance of these controllers through extensive experimentation. This experimentation includes hundreds of hours of actual flight experiments and over a thousand hours of simulation.

The roll, pitch, and collective controllers specified in Garcia[14] utilize positional error as input for determining control. To allow these controllers the ability to interface with the formation control algorithm's vector outputs, an adapter function was implemented. This adapter function is designed to translate the local unit vector of the potential fields into axis specific positional error for direct input into the fuzzy controller. Conversion is done via a constant multiplier that is tuned to achieve a desired level of aggression for the flight maneuvers as well as position accuracy.

In addition to the unmanned systems, the manned vehicle is also controlled via fuzzy logic. Utilizing an automated control system had two main benefits. First the authors were able to perform experiments without the need for a heavily trained pilot. Second, the authors were able to create consistent experiments and thus keep any pilot inconsistencies from effecting simulation results. Fuzzy controllers were utilized simply due to their availability but any sufficient vehicle controller would suffice. The specific details of the fuzzy controllers used to stabilize and navigate the manned vehicle were developed from controllers found in Garcia et al.[13] These controllers were modified to include a larger rule set and simplified membership functions.

2.5 Splinter Group Surveillance

In addition to accompanying the manned vehicle in formation, the UASs are able to splinter off and survey areas of interest, or hot spots. Hot spots may be persistent or temporary and the level of interest in a specific hot spot may vary over time. Let (x_h, y_h, z_h) be the location of a hot spot and let the vector field associated with this hot spot be

$$H\begin{pmatrix} x \\ y \\ z \end{pmatrix} = W_H(x, y, z) \begin{pmatrix} x - x_H \\ y - y_H \\ z - z_H \end{pmatrix} \tag{7}$$

In this discussion, hot spots are pre-determined, and interest in a particular hot spot depends on the proximity of the manned vehicles to that hot spot. The parameter $d_H(x,y,z)$

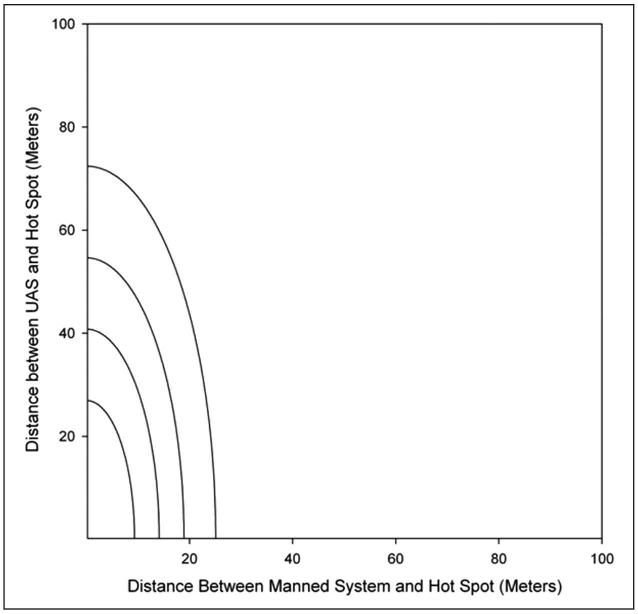

Figure 2. Contour plot of WH(x,y) as a function of the UAS/Hot Spot distance and the Manned System/Hot Spot distance.

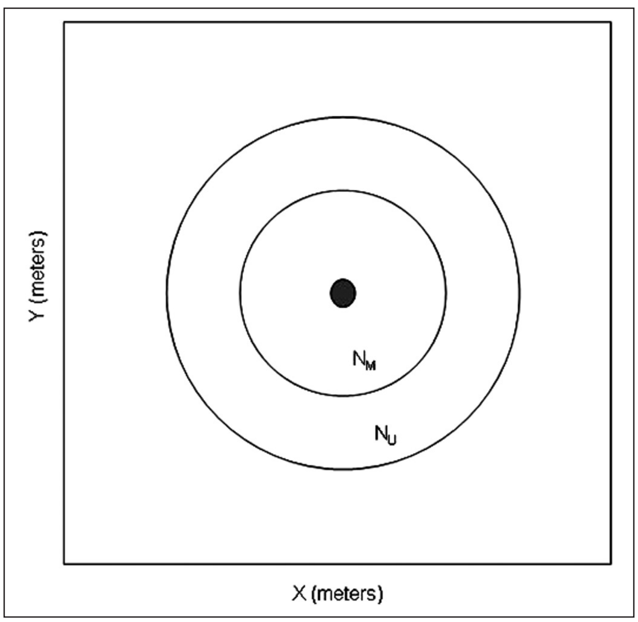

Figure 3. The nested N_M and N_U neighborhoods centered around the Hot Spot.

is defined as the square of the distance between any point (x,y,z) and the hot spot:

$$d_H(x,y,z) = (x - x_H)^2 + (y - y_H)^2 + (z - z_H)^2 \quad (8)$$

The weighting function

$$W_H(x,y,z) = e^{-\alpha_H d_H(x,y,z)} e^{-\alpha_M d_H(x_M,y_M,z_M)} \quad (9)$$

shown in Figure 2, attracts nearby UASs provided that the manned vehicles are in close proximity to that hot spot. The UASs will leave the hot spot once the manned vehicles' proximity has reached a safe distance. The parameter $\alpha_H > 0$ controls the size of the region of attraction around the hot spot. The parameter $\alpha_m > 0$ is based on a safety distance, *sd*, or radius of influence for the hot spot. If $d_H(x_m,y_m,z_m) < sd$, the UASs will continue to monitor the hotspot. Otherwise, they will rejoin the manned vehicle in formation. Figure 3 illustrates this point. If the manned system is inside the neighborhood N_M, then UASs within the neighborhood N_U are attracted to the Hot Spot. UASs outside N_U are too far away to be attracted to the Hot Spot regardless of the location of the manned system. By combining splinter group surveillance with the formation control and obstacle avoidance vectors from (1) and (4), the overall motion of the team is created:

$$\begin{bmatrix} v_{x_{tot}} \\ v_{y_{tot}} \\ v_{z_{tot}} \end{bmatrix} = \begin{bmatrix} v_x + v_{x_{\text{avoid}}} + H_x \\ v_y + v_{y_{\text{avoid}}} + H_y \\ v_z + v_{z_{\text{avoid}}} + H_z \end{bmatrix} \quad (10)$$

3. Intent Prediction

In addition to dynamic formations and splinter groups, several coordinated flight issues are addressed. Formation control is accomplished by utilizing team members' physical locations to coordinate motion. This particular implementation has a weakness due to delays between control input and vehicle response. This delay causes formation deformation during flight maneuvers. This becomes visible during basic maneuvers such as forward flight where the formation begins to stretch over time, see Figure 4. Deformations were further exaggerated due to the lack of global information. Since team members are only aware of the state of the vehicle they are immediately following, the motion delay is propagated between the manned vehicle and each subsequent vehicle. Consequently, maximum deformation is a function of the number of vehicles in the team, the formation type, and the input to response delay.

The origin of this issue was the lack of knowledge about the actions of the other vehicles in the formation. To mitigate this issue, an intention prediction algorithm was implemented. Intent prediction, for the purpose of this work, refers to predicting the future velocity vector of another vehicle within sensing range. Although the optimal solution for prediction would utilize a vehicle-specific mathematical model, this work utilizes a fuzzy reasoning system to show that a more general solution can be achieved. Fuzzy logic was chosen due to its ability to cover a wider range of operating conditions, its utilization of natural language, and its ability to gracefully handle incorrect or conflicting input data. One desirable feature of utilizing a generalized solution is that it provides some level of confidence given a pairing with unknown, incorrectly modeled, or damaged

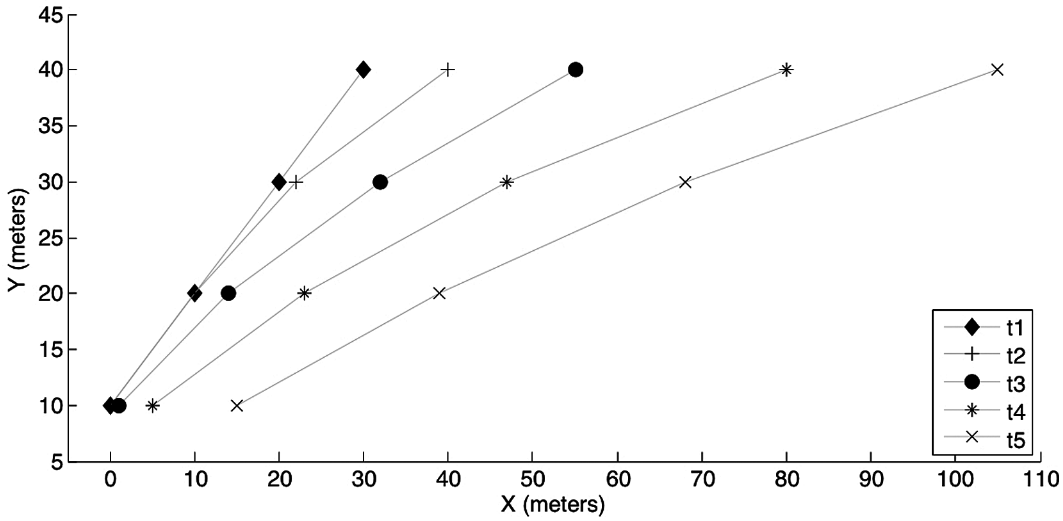

Figure 4. Example of the type of deformation caused by control/response delays.

Table 1. Description of fuzzy flight path prediction controller.

Inputs	Range (units)	# MFs	Function #1	Function #2	Function #3
Velocity (ft/s)	–10 to 10 (ft)	3	Trapezoidal [–1000,–1000,–5,0]	Normal (std_dev = 2, mean = 0)	Trapezoidal [1000,1000,5,0]
Angle (degrees)	–25 to 25 (deg)	3	Trapezoidal [–1000,–1000,–10,0]	Normal (std_dev = 5, mean = 0)	Trapezoidal [1000,1000,10,0]
Angular Rate (degrees/s)	–100 to 100 (deg/s)	3	Trapezoidal [–1000,–1000,–40,0]	Normal (std_dev = 20, mean = 0)	Trapezoidal [1000,1000,40,0]

vehicles. Although it is not discussed in this work, generalized control solutions can be tuned via online algorithms and can provide a near optimal solutions that are robust to the unknown.[16]

The fuzzy reasoning system utilizes the lead vehicle's state data as input. This state data includes pose, angular rate, and the velocity vector. This state data was selected because it provides a good identifier of the desired direction of travel for rotary wing vehicles and can be estimated from a remote location using various sensors.[17,18]

The fuzzy reasoning system is made up of two fuzzy inference engines for roll and pitch, respectively. Each inference engine takes as input a vehicle's estimated velocity, orientation, and angular rate for a given axis. Using this data, the inference engine attempts to predict the strength and direction of the desired movement along that axis. The values returned by the inference engines are in the range [–1, 1] representing magnitude and direction along that axis. These values are then combined to form a 2D vector representing the desired lateral and longitudinal path of travel.

Both the roll and pitch inference engines are identical and simply take in values corresponding to data along that axis. The inference engines are Sugeno-type constant inference engines utilizing a weighted average defuzzification method and 27 rules. Table 1 details the exact makeup of the fuzzy inference engines, and Table 2 details the rules of the pitch axis inference engine. The constant outputs associated with the controllers are –1, 0, and 1 corresponding to backward, stationary, and forward, respectively. The roll inference engine has the exact same rules with differing labels corresponding to that axis.

A second issue arises when the manned vehicle, executing an evasive maneuver, aggressively pursues a path in the direction of the formation. The navigation approach discussed in Section 2 is not sufficient for this case – the manned vehicle would quickly overtake and collide with the unmanned vehicle. This issue is the result of vehicle and coordination limitations. First, the heterogeneous implementation created a team with differing flight capabilities. Unmanned members are more agile than manned members but the manned vehicles can achieve higher speeds than the unmanned vehicles. Second, team coordination algorithms are designed to hold formation regardless of the situation. Since the unmanned vehicle's goal vector and obstacle

Table 2. Fuzzy rules for the predicting flight path along the pitch axis.

If	Velocity is:	&	Angle is:	&	Angular Rate is:	then	Predicted Path is:
	Backward		Backward		Backward		Backward
	Backward		Backward		Small		Backward
	Backward		Backward		Forward		Stationary
	Backward		Small		Backward		Backward
	Backward		Small		Small		Backward
	Backward		Small		Forward		Stationary
	Backward		Forward		Backward		Stationary
	Backward		Forward		Small		Stationary
	Backward		Forward		Forward		Stationary
	Small		Backward		Backward		Backward
	Small		Backward		Small		Backward
	Small		Backward		Forward		Stationary
	Small		Small		Backward		Backward
	Small		Small		Small		Stationary
	Small		Small		Forward		Forward
	Small		Forward		Backward		Stationary
	Small		Forward		Small		Forward
	Small		Forward		Forward		Forward
	Forward		Backward		Backward		Stationary
	Forward		Backward		Small		Stationary
	Forward		Backward		Forward		Stationary
	Forward		Small		Backward		Stationary
	Forward		Small		Small		Forward
	Forward		Small		Forward		Forward
	Forward		Forward		Backward		Stationary
	Forward		Forward		Small		Forward
	Forward		Forward		Forward		Forward

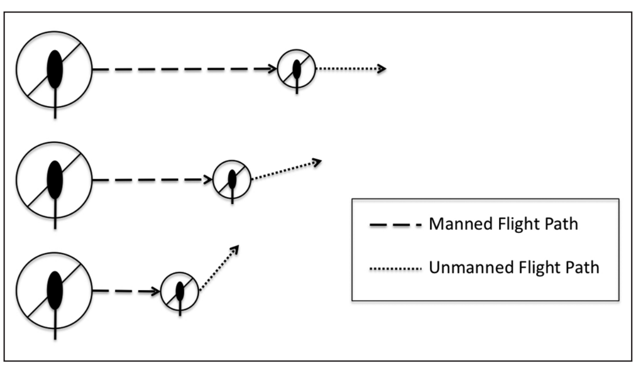

Figure 5. Example of flight path deformation based on overtaking vehicle.

avoidance vector are in the same direction as the aggressor vehicle's flight path, the unmanned vehicle never moves out of the way.

The root of this issue is also caused by a lack of knowledge about what other vehicles are planning to do. Since it is unrealistic to model a manned–unmanned team as a single homogeneous group, intent prediction algorithms are necessary. To mitigate this issue, unmanned vehicles can deviate from formation in extreme circumstances. This algorithm was implemented by altering the repulsive obstacle avoidance vector, discussed in Section 2.2, of the aggressor vehicle based on relative distance and the predicted velocity vector. Alteration of the repulsive vector was performed by rotating the vector towards the perpendicular of the aggressors' predicted path. The extent of this rotation towards perpendicular is a function of both the predicted velocity and the distance between the two vehicles. As a result, the vehicle in the flight path of an overtaking vehicle will ultimately choose a flight path more perpendicular to that of the overtaking vehicle, see Figure 5.

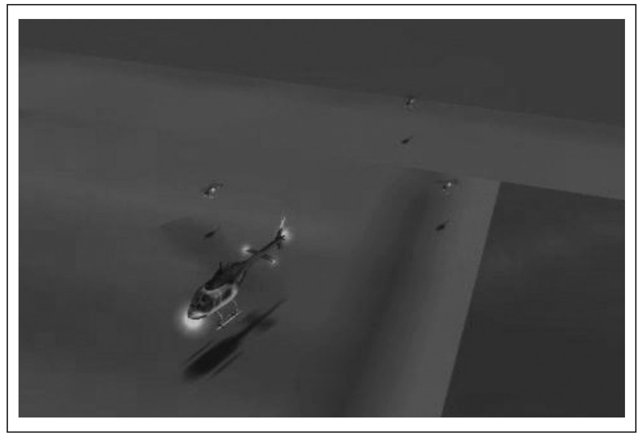

Figure 6. Snapshot of a staggered trail formation in the X-Plane environment.

4. Flight Tests

Flight test were performed using a multi-UAS simulator system described in Garcia and Barnes.[19] This simulation system utilizes the commercial simulator X-Plane coupled with a small cluster of dedicated computers. By utilizing this system users are able to experiment with control of multiple aircraft simultaneously in an environment that has proven world and aircraft models.[20] Communication and coordination with the individual X-Plane simulations is performed using a custom Matlab/Simulink model. This model contains the individual vehicle controllers as well as the integrated formation control algorithms. A screen shot from the X-Plane environment is shown in Figure 6.

In order to demonstrate the proposed approach and validate the methodology, numerous flight tests were performed. In these experiments a manned vehicle is given a set of waypoints to traverse. The manned vehicle is then teamed with a group of three UAS systems that are tasked to travel with the manned aircraft in various formations while simultaneously performing surveillance of hot spots.

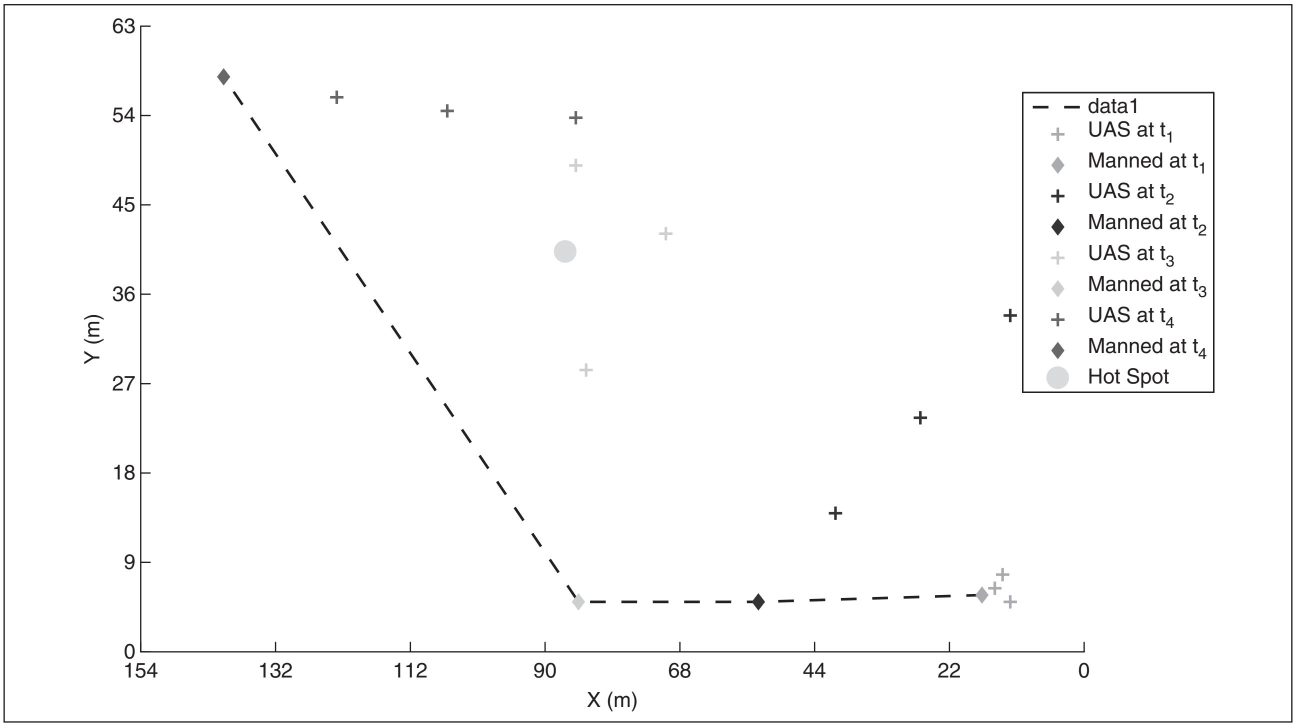

Figure 7. Snapshot of manned vehicle mission with three UASs at different time slices where t_1=1, t_2=200, t_3=300, and t_4=600.

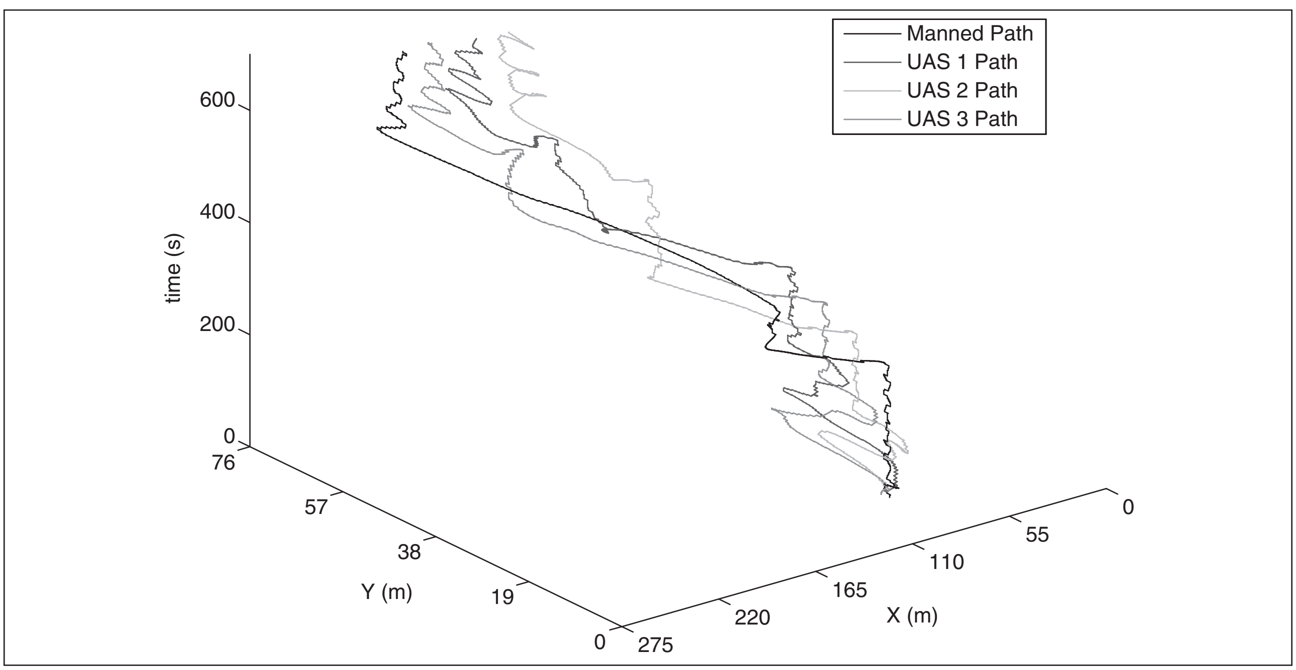

Figure 8. Manned and unmanned system paths with time on the z-axis.

4.1 Splinter Group Surveillance

In one set of flight tests, UASs were commanded to follow the manned aircraft in a right echelon formation. In these tests, there is a single hot spot located within the environment. Figure 7 shows snapshots of the manned and unmanned vehicles at various time slices. The UASs maintain the right echelon formation until the manned vehicle enters a close proximity with the hot spot. At this time, the UASs diverge from formation to explore the hot spot. At time slice t_3, the UASs surround the hot spot. When the manned system has reached a safe proximity from the hot spot, the UASs rejoin

the manned vehicle in formation. The flight paths associated with the snapshots in Figure 7 are detailed in Figure 8 to show where the UASs diverge from and rejoin formation.

4.2 Dynamic Formation Change from Right Echelon to Staggered Trail

In the second set of flight tests, UASs were commanded to follow the manned aircraft in a right echelon formation. During the flight the UASs were re-tasked to form a staggered trail formation. Figure 7 shows snapshots of the manned and unmanned systems at various time slices. The UASs maintain the right echelon formation until commanded to modify between t_3 and t_4. At this time, the UASs break apart to form a staggered trail formation. At time slice t_5, the UASs complete the requested formation change. The flight paths associated with the snapshots in Figure 9 are detailed in Figure 10 to show where the UASs modify their formation.

4.3 Intended Flight Path Prediction

To determine the effects of flight path prediction on the team formation direct comparisons were made between position only formation control and path prediction formation control. As shown in Figure 11, by including intended path

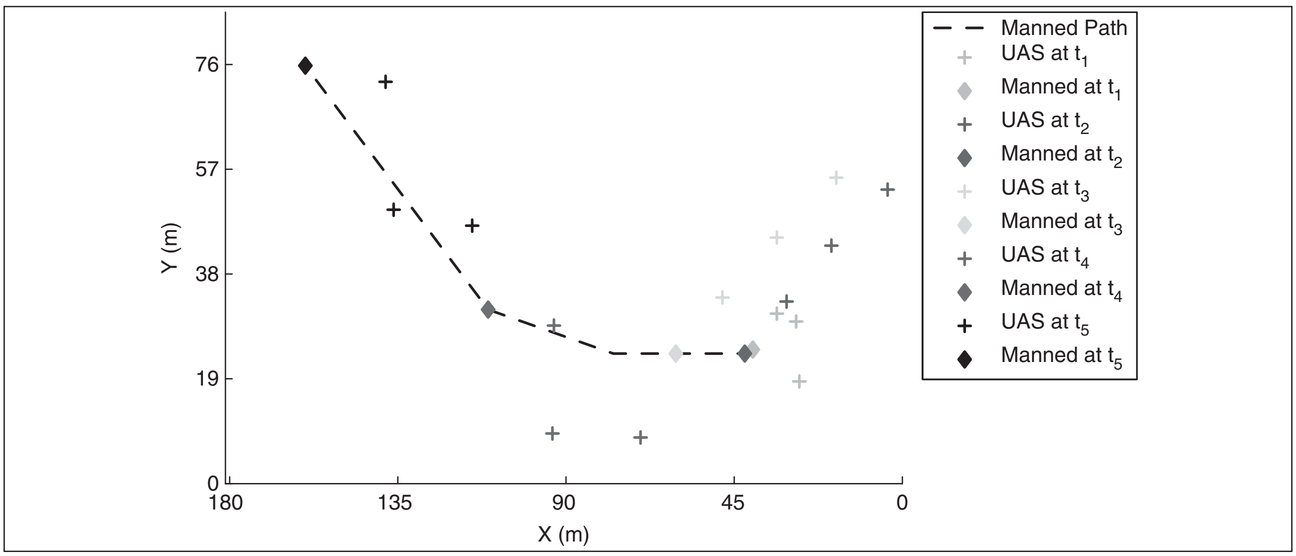

Figure 9. Snapshot of manned vehicle mission with three UAS at different time slices where t_1=1, t_2=100, t_3=200, t_4=300, and t_5=400.

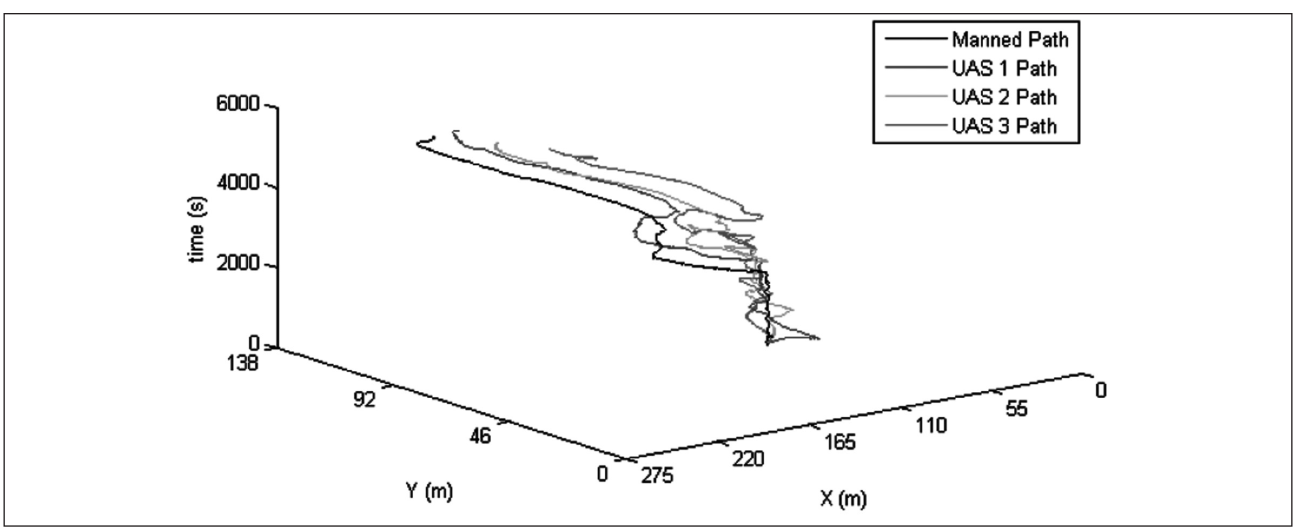

Figure 10. Manned and unmanned system paths with time on the z-axis.

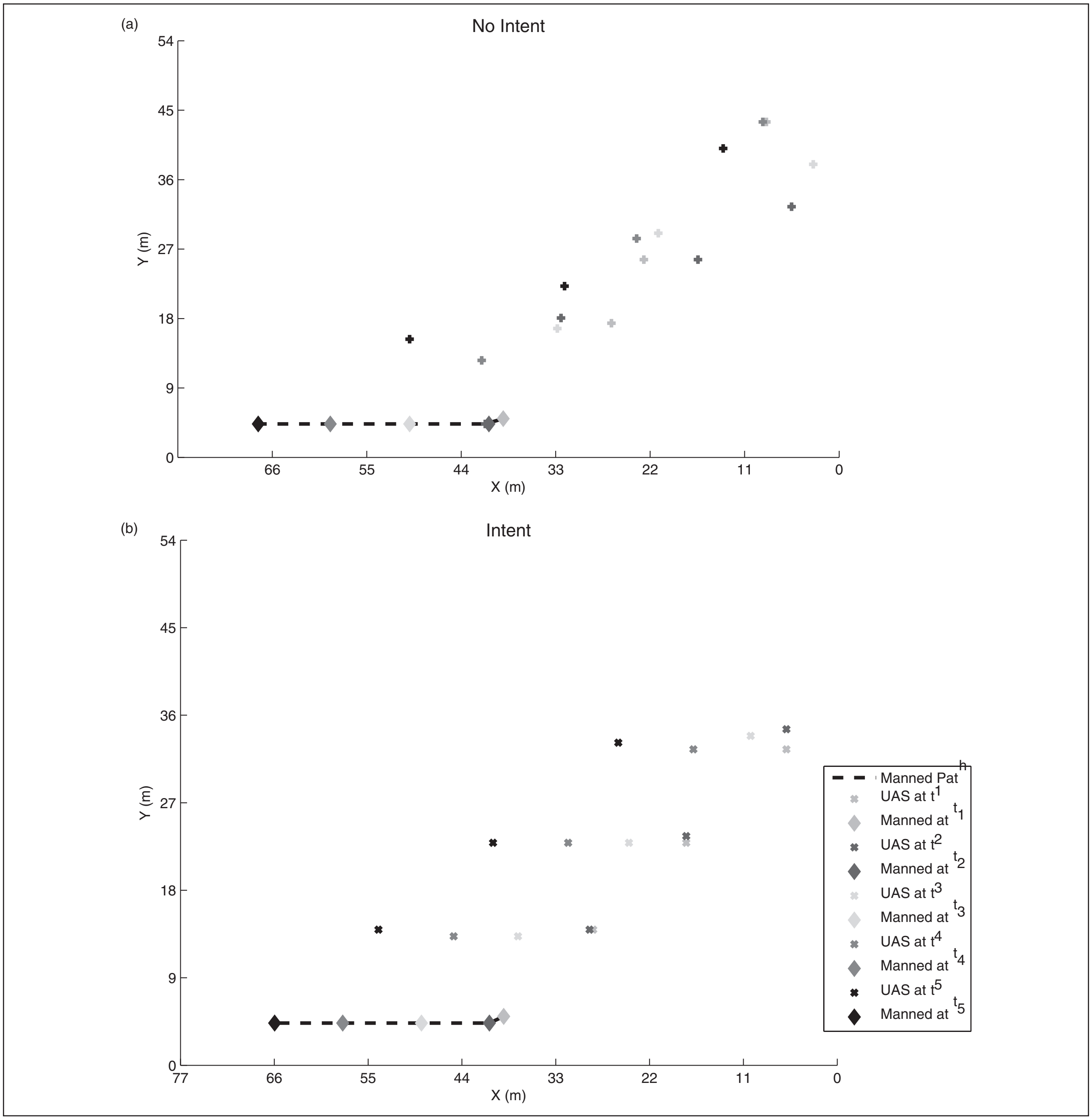

Figure 11. Snapshot of team movement with (a) no intent and with (b) intent prediction (t_1=120, t_2=150, t_3=160, t_4=170, and t_4=180).

prediction, the team was able to hold a tighter and more controlled formation from a standstill to a maneuver. This increased accuracy allows the team to function in a closer proximity while maintaining a level of safety.

5. Conclusions

This paper introduces preliminary work being performed in support of the ARL UASs as Wingmen project. This work specifically focused on methods for achieving automatic and self-assigned standard flight formations, breakaway splinter groups, and intended flight path prediction. Simulation experiments were provided in support of the claims made throughout the paper.

The work presented here details initial solutions to only a select few of the numerous issues involved in integrating unmanned and manned aircrafts into functional and safe teams. Issues such as perception, failure tolerance, communication, task assignment, task prioritization, and emergency maneuvers will also require extensive advancement before

the technology has matured to functional implementation. Future work on this project includes online adaptation of the flight controllers, automated formation selection, and intelligent evasive maneuvers.

Funding

This work was supported and partially funded by the Vehicles Technology Directorate (VTD-UV) of the Army Research Laboratory located at Aberdeen Proving Ground, MD.

Conflict of interest statement

None declared.

6. References

1. Merino, L.; Caballero, F.; Martinez-de Dios, J.R. Ollero, A. Cooperative Fire Detection using Unmanned Aerial Vehicles, Robotics and Automation, 2005. ICRA 2005. *Proceedings of the 2005 IEEE International Conference on*, pp. 1884–1889, 18–22 April 2005: doi: 10.1109/ROBOT.2005.1570388.
2. Altshuler Y, Yanovsky V, Wagner I and Bruckstein A. Efficient cooperative search of smart targets using uav swarms. *Robotica* 2008; 26: 551–557.
3. Shima T, Rasmussen S and Gross D. Assigning micro UAVs to task tours in an urban terrain. *IEEE Trans Control Syst Technol* 2007; 15: 601–612.
4. Puri, A, Valavanis, AR, Kontitsis, M. Statistical profile generation for traffic monitoring using real-time UAV based video data. *Control & Automation*, 2007. MED '07. Mediterranean Conference pp. 1–6, 27–29 June 2007.
5. Girard, AR, Howell, AS, Hedrick, JK. Border patrol and surveillance missions using multiple unmanned air vehicles. *Decision and Control*, 2004. CDC. 43rd *IEEE Conference on, vol.1*, pp. 620–625 Vol.1, 17–17 Dec. 2004.
6. Ackerman S. Air Force wants drones to sense other planes' 'Intent'. http://www.wired.com/dangerroom/2010/07/air-force-wants-drones-to-sense-other-planes-intent/ *Wired* 2010.
7. United States Air Force: Unmanned Aircraft Systems Flight Plan 2009-2047. Washington DC, 18 May 2009.
8. Hart DM, Craig-Hart PA. Reducing swarming theory to practice for UAV control. *Aerospace Conference*, 2004. *Proceedings. 2004 IEEE , vol.5*, pp. 3050–3063 Vol.5, 6–13 March 2004 doi: 10.1109/AERO.2004.1368111.
9. Waydo S, Hauser J, Bailey R, Klavins E and Murray RM. UAV as a reliable wingman: A flight demonstration. *IEEE Trans Control Syst Technol* 2007; 15: 680–688.
10. Garcia RD and Valavanis KP. The implementation of an autonomous helicopter testbed. *J Intell Robotics Syst* 2009; 54: 423–454.
11. Barnes L, Fields MA and Valavanis K. Swarm formation control utilizing elliptical surfaces and limiting functions. *IEEE Trans Syst, Man, Cyber B* 2009; 39: 1434–1445.
12. Fields MA, Haas E, Hill S, Stachowiak C and Barnes L. Effective robot team control methodologies for battlefield applications. *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference*, pp. 5862–5867, 10–15 Oct. 2009.
13. Garcia RD, Valavanis KP and Kandel A. Autonomous helicopter navigation during a tail rotor failure utilizing fuzzy logic. *Control & Automation*, 2007. MED '07. Mediterranean Conference, pp. 1–6, 27–29 June 2007.
14. Garcia RD. Designing an autonomous helicopter testbed: from conception through implementation. *Computer science and engineering*. Tampa: University of South Florida, 2008.
15. HLZ Handout. USAP School, Department of the Army, 2009.
16. Gao Z, Trautzsch TA and Dawson JG. A stable self-tuning fuzzy logic control system for industrial temperature regulation. *IEEE Trans Ind Applicat* 2002; 38: 414–424.
17. Taati B and Greenspan M. Satellite pose acquisition and tracking with variable dimensional local shape descriptors. *IEEE/RSJ IROS Workshop on Robot Vision for Space Applications*, 2005.
18. Jasiobedzki P, Greenspan M and Roth G. Pose determination and tracking for autonomous satellite capture. *International Symposium on Artificial Intelligence, Robotics, and Automation in Space*, 2001. Montréal, Québec, Canada; The 6th International Symposium on Artificial Intelligence, Robotics and Automation in Space
19. Garcia R and Barnes L. A multi-UAV simulator utilizing X-Plane. *J Intell Robot Syst* 2010; 57: 393–406.
20. FAA – Certified X-Plane, http://www.x-plane.com/pg_certified.html.

Biographies

Richard D Garcia received his BS from Texas Tech University and his MS and PhD from the University of South Florida. He has worked for IBM, Lockheed Martin, and as a contractor to the Army Research Laboratory. His main research areas are in Unmanned Systems with a focus on intelligent control and decision-making.

Laura E Barnes received a BS in computer science from Texas Tech University in 2003. She received MS and PhD. degrees in computer science and engineering from the University of South Florida, Tampa, FL, in 2007 and 2008, respectively. From 2003 to 2008, she was a research assistant in the Department of Computer Science in the University of South Florida performing research in the areas of human–robot interaction, unmanned systems, and swarm control. In addition, she held a fellowship from the Army Research Laboratory and performed research in the Vehicles Technology Directorate. From June 2008 to May 2010, she was a faculty member in the Automation and Robotics Research Institute at the University of Texas Arlington where she performed research in swarm robotics and microsystems. In May 2010, she joined the University of South Florida as an Assistant Professor. Her primary research interests include robotics, intelligent systems, and decision support systems.

MaryAnne Fields received a PhD degree in mathematics from Clemson University, Clemson, SC, in 1989. She is the Team Leader for the Intelligent Control team of the Unmanned Systems Division of the Vehicles Technologies Directorate. Her recent research interests include planning for uncertain environments, tactical behavior algorithms for teams of robots, and use of simulation tools and virtual environments for both development and testing of robotic systems.